Schedae Informaticae Vol. 24 (2015): 159–177
doi: 10.4467/20838476SI.15.015.3997

Evolutionary Algorithm for Particle Trajectory Reconstruction within Inhomogeneous Magnetic Field in the NA61/SHINE Experiment at CERN SPS

Oskar Wyszyński[1,2]
[1]Faculty of Physics, Astronomy and Applied Computer Science
Jagiellonian University, Łojasiewicza 11, 30-348 Kraków, Poland
[2]CERN, CH-1211 Geneva 23, Switzerland
e-mail: *oskar.wyszynski@cern.ch*

Abstract. In this paper, a novel probabilistic tracking method is proposed. It combines two competing models: (i) a discriminative one for background classification; and (ii) a generative one as a track model. The model competition, along with a combinatorial data association, shows good signal and background noise separation. Furthermore, a stochastic and derivative-free method is used for parameter optimization by means of the Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES). Finally, the applicability and performance of the particle trajectories reconstruction are shown. The algorithm is developed for NA61/SHINE data reconstruction purpose and therefore the method was tested on simulation data of the NA61/SHINE experiment.

Keywords: tracking, event, reconstruction, particle, high, energy, physics, HEP, NA61, SHINE, CERN, TPC, magnetic, field, CMA, evolutionary, strategy, bayes, generative, discriminative.

1. Introduction

In the field of High Energy Physics (HEP), reconstruction of charged particle trajectories is a challenging task in terms of efficiency and computational complexity. The first commonly used techniques were histogramming and Hough transformation [1]. Unfortunately, histogramming shows poor efficiency, when dealing with high density of tracks and relatively high measurement uncertainty, whereas the Hough Transformation is characterized by high computational complexity compared to its efficiency.

The flaws of employed methods became more visible during the quest for scientific discoveries which lead to higher energies, implying higher multiplicity of particles. In late 90's the Kalman filter was introduced to event[1] reconstruction in HEP by Billoir [2–4] and Fruhwirth [5], becoming the most popular algorithm for particle trajectory reconstruction.

The Kalman filter [6] lies under most of track reconstruction algorithms used in modern experiments, including experiments located at the Large Hadron Collider (LHC) [7]. The LHC, located at CERN in Geneva, is the world's largest and most powerful particle accelerator. All four LHC experiments based their algorithms on the Kalman filter with great success: ATLAS [8,9], CMS [10–12], ALICE [13,14] and LHCb [15–17].

Despite of the success of Kalman filter, there are parallel efforts to develop new tracking algorithms which use much simpler models, and at the same time provide similar performance without using an approximation. Often, creation of the state transition, control and observation matrices is difficult, especially when dealing with multidimensional data. Furthermore, a desired feature is to eliminate slow combinatorial algorithms for newly included particle detectors in the system.

In order to simplify such excessive processes, a search for a more general algorithm was performed. The field of Machine Learning turned out to be a natural choice. For reference, many techniques were studied, namely the Cellular Automaton for the LHCb Outer Tracker [18] as well as a recursive neural network known as Hopfield Network in the LHCb Muon System [19], developed in the LHCb experiment. Another experiment, ALICE, presents an applicability of the Denby-Peterson network for the Inner Tracking System [20–22]. Those new methods are often accompanied by the Kalman Filter[18, 20–22]. Applying a search for potential points for extension of track candidates, these methods preserve the drawback of Kalman filter. A detailed overview of the state of event reconstruction techniques in HEP is provided in [23] and [1].

In this paper, a novel evolutionary algorithm is presented along with application on particle trajectory reconstruction in the NA61/SHINE experiment, located at the CERN SPS accelerator. The facility of NA61/SHINE experiment [24] consists of many sub-detectors, where the key componets are the Time Projection Chambers (TPC) [25] shown in Fig. 1. Vertex TPCs are considered problematic due to the presence of the magnetic field and very high track densities. Therefore, our study is focused on the vertex TPC local track reconstruction.

2. The raw data

In order to understand the algorithm, the mechanism of acquiring the data has to be presented. First, we will describe detector working principles and the raw data output. Second, the clusterization algorithm will be explained, which is the last processing

[1] An event refers to the results of a single fundamental interaction, which took place between subatomic particles.

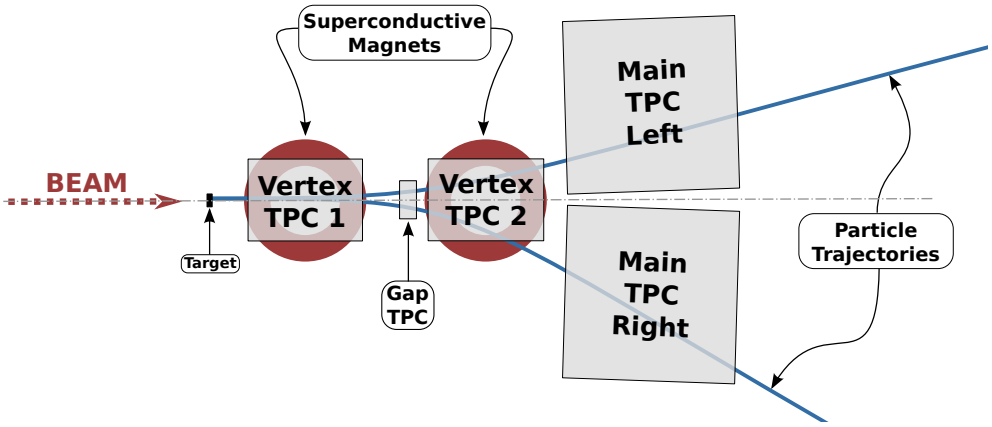

Figure 1. Simplified detector configuration of the NA61/SHINE facility. The beam, consisting of particles accelerated to relativistic speeds, is projected on a fixed target. The produced particles traverse the Time Projection Chambers (TPCs), where their trajectories are bent by the $\sim 1.5T$ magnetic field of the superconducting spectrometer magnets.

step before tracking. Finally, the nature of the particle trajectories is briefly described in order to define the problem which needs to be solved by the presented algorithm.

2.1. Time Projection Chambers

A Time Projection Chamber (TPC), indicated in Figure 1, consists of particle detectors placed inside a volume of noble gas located in a homogeneous electric field. A particle passing within the TPC sensitive volume ionizes the gas. The resulting primary ionization electrons drift toward the amplification-and-readout plane due to the electric field, as show in Figure 2. The amplification-and-readout plane, containing an array of readout pads equipped with an amplifier, a time sampler and charge digitizer electronics, provides the digital data about the ionization trace. The Z coordinate along the track is parameterized by the padrow location index. At a given padrow, the X value and Y value are derived from the charge deposit profile of the track intersection with the padrow: the X coordinate is determined by the charge deposition intensity along pad index, whereas the Y coordinate (drift depth) is determined from the charge deposition profile in terms of time slice index, after Detector/clusterization. The described structure of raw data is shown in Figure 3. A number of detector effects can slightly influence the shape of a measured charge deposition cluster of a track intersection with a padrow. For instance, particles traversing the bottom part of the detector, and thus having a larger drift path, produce clusters with a larger radius due to charge cloud diffusion in the working gas. This implies a deterministic distortion effect on the signal. Fluctuations of the ionization process along with the electronic

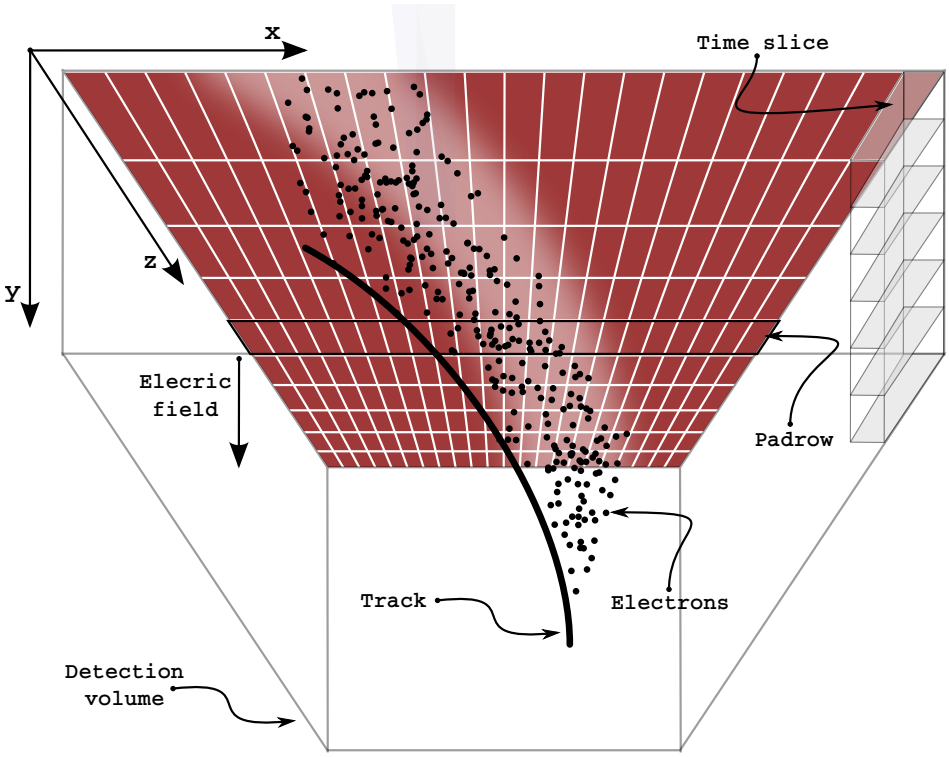

Figure 2. Simplified illustration of TPC working principle.

pickup noise on the readout pads, on the other hand, superimpose a statistical uncertainty on the measured charge depositions. The ensemble of all such detector effects complicates the task of track pattern recognition in the system. Further readings on the data readout is provided by [26] and [27].

2.2. Clusterization of ADC signals on a padrow

The number of deposited charge ADC signals within one event amounts to about 5×10^7 regardless what type of reaction is considered. It is a considerable amount of data, especially in a situation where millions of events per reaction are collected. Therefore, reduction of the data is needed by means of charge clusterization on each padrow. This approach leads to significant reduction of the data size (factor of around thousand for low particle multiplicity events). We aimed at having a fast clusterization and therefore used a simple algorithm with value based start and stop criteria and subsequent quality assessment of parameters such as shape, maximum and average values of ADC signals. In the end, we produce the clusters with a point in the 3D space as the main attribute.

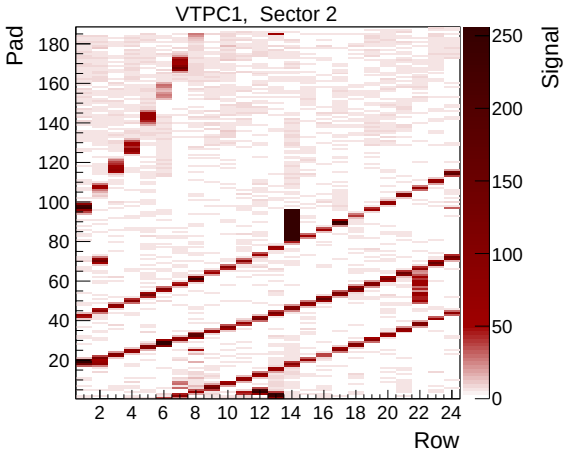

Figure 3. Maximal value of digitalized charge within one time slice (ADC signal) measured for every pad in second sector of vertex TPC 1. One can recognize trajectories produced by four traversing particles.

2.3. Particle motion

Recognition of particle trajectories in chambers without presence of magnetic field such as MTPCs, reduces the problem to finding straight lines. However, when magnetic field is applied, the equation of motion has to incorporate the deflection caused by the Lorentz force F_l

$$F_l = \frac{d\mathbf{p}}{dt} = q \cdot (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \approx q \cdot \mathbf{v} \times \mathbf{B} \tag{1}$$

In case of NA61/SHINE, the strength of the electric field in the TPCs is of the order of $\sim 20kV/m$, whereas the magnetic field force is approximately $\sim 3 \cdot 10^8 V/m$. Therefore, the neglection of electric field is justified. The elapsed time t along the track flight can be reparameterized using flight path length s as $dt = ds/v$, where $v = |\mathbf{v}|$ is the magnitude of velocity vector. Introducing $p = |\mathbf{p}|$ for the momentum magnitude, the unit vector $\mathbf{e} = \mathbf{v}/v = \mathbf{p}/p$ gives us the following equation

$$d\mathbf{e} = \frac{q}{p} \cdot \mathbf{e} \times \mathbf{B} \cdot ds \tag{2}$$

Afterwards, using a simple but tedious mathematical transformation, the path length s is reparameterized by detector coordinate z, see for instance [28]. This leads to a system of differential equations, z being the running parameter, resulting in the following extrapolation operator

$$\mathcal{Q} : T \times \mathbb{R} \to T, \tag{3}$$

where T is a track parameters vector space. Thus, an outcome of the $\mathcal{Q}$, given a track parameter vector Θ_0 and destination z, is equal to

$$\hat{\Theta} = \mathcal{Q}(\Theta_0, z). \tag{4}$$

Often instead of the above equation, an approximation is used, e.g. a helix track. That is only possible in situations where the magnetic field is relatively homogeneous. However, to acquire a homogeneous magnetic field in practice is a challenging task, and therefore the helix equation may not always be a good approximation to the true solution of our differential equation. Nevertheless, the helix approximation may be handy for rough track parametrization in the parts of an algorithm where great accuracy is not required.

3. Evolutionary tracker

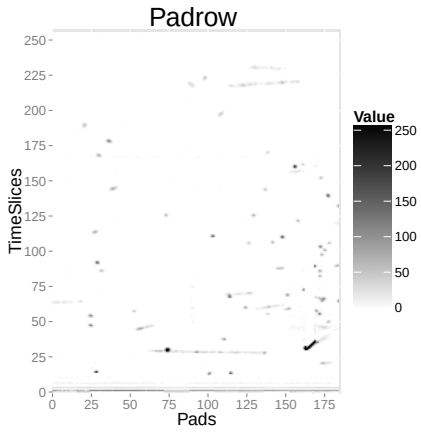

Figure 4. Values of digitized charge for every pad and every time slice for an example padrow. The clusters (circular spots), determine locations where particles traversed the particular padrow.

The evolutionary tracker is a recursive algorithm that uses a series of measurements observed over an elapsing parameter such as time or z coordinate in our case, and produces estimates of a track parameters Θ. The algorithm involves two stages: prediction and measurement update. A prediction is made by extrapolating a track parameter vector with equation (3) to a Z position of a next measurement, namely a cluster as in Section 2.2.

The proposed algorithm consists of three principal components. (i) Model of an event – a whole event is modeled using a discriminative model of a background and a generative model of a particle trajectory. Competition between those models shows good signal to noise separation. (ii) Measurement-to-track association – this is performed using breadth search along with sorting of the measurement in order to construct time-line. (iii) Continuous track parameter optimization – is realized by means of the Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES), stochastic and derivative-free method proved to be working for an objective ill-conditioned function with discontinuities where other derivative based and Quasi-Newton methods failed. In this section each of the components are described.

3.1. Model of an event

The event model consists of the two competing models, the generative model of a track and discriminative model of a background. The aim of the competition is an efficient method to separate signal from noise. In this section both models are described along with a reason why the generative model was used for a track, whereas discriminative models shows, in general, better performance [29].

A prior probability is estimated from simulated training samples and a likelihood function is constructed based on known detector resolution. A one variable logistic regression was used for the background, which proved to be sufficient.

3.1.1. Naïve Bayes trajectory model

The Naïve Bayes trajectory model assumes all parameters to be independent and identically distributed as the track parameters are. Interestingly, the Naïve Bayes demonstrates good performance in practice [30], also for problems which violate the assumption of statistical independence. The explanation of this phenomenon can be found in [31], which shows that the optimality of Naïve Bayes approach does not depend on the independence attribute and the applicability is much greater than the original restrictive assumptions would suggest.

The main reason of choosing a generative model is the fact that uncertainty of track cluster position estimates are well known, hence there is no need to estimate it from scarce labeled data. With constant arbitrary likelihood parameters, only a prior probability estimate is required. Therefore, the Naïve Bayes model was chosen although the the logistic regression is expected to overtake the performance of the Naïve Bayes method as the number of training samples is increased [29].

The particle trajectory model is described as

$$\overbrace{P(\hat{\mathbf{\Theta}}|\mathcal{S}(\mathbf{c}))}^{\text{Posterior probability}} \propto \underbrace{P(\hat{\mathbf{\Theta}})}_{\text{Prior probability}} \overbrace{P(\mathcal{S}(\mathbf{c})|\hat{\mathbf{\Theta}})}^{\text{Likelihood}} \tag{5}$$

where $\mathbf{c} = (x, y, z, s_1, \ldots, s_n)$ stands for a cluster with center of gravity located at (x, y, z) along with charge deposition ADC signals $(s_1, \ldots, s_n)$ in the cluster. The symbol $\hat{\mathbf{\Theta}}$ denotes an estimation for a track parameter vector $\mathbf{\Theta} = p \oplus o$, where $p = (p_x, p_y, p_z)$ denotes the particle momentum vector at a starting point $o = (x, y, z)$ and $\oplus$ denotes concatenation operator ($dim(\mathbf{\Theta}) = dim(p) + dim(o)$).

The operator $\mathcal{S} : T \cup C \to \mathbb{R}^3$ produces a three dimensional Euclidean space vector

$$\mathcal{S}(\xi) = (\xi_x, \xi_y, \xi_z) \tag{6}$$

where C is a cluster parameter vector space. The model (5) consists of two components: likelihood function of a cluster, given a track; and an *a priori* probability of a particular track parameter vector. The likelihood is defined in the following manner:

$$P(\mathcal{S}(\mathbf{c})|\hat{\mathbf{\Theta}}) = P(\mathcal{S}(\mathbf{c})|\hat{\mathbf{\Theta}}, \Sigma) \sim \mathcal{N}(\hat{\mathbf{\Theta}}, \Sigma) \tag{7}$$

where '$\sim$' denotes equality in distribution, $\hat{\mathbf{\Theta}}$ is an estimate of parameters $\mathbf{\Theta}$ being extrapolated by equation (3) to a position $\mathbf{c}_z = \hat{\mathbf{\Theta}}_z$ and $\mathbf{\Sigma}$ is a diagonal covariance matrix. The parameters of the likelihood function are contained in the diagonal covariance matrix

$$\mathbf{\Sigma} = diag\{\sigma_x^2, \sigma_y^2, \sigma_z^2\} \tag{8}$$

Note that the value of σ_z^2 is irrelevant as z position of a cluster is not a stochastic variable, but is determined by the padrow location.

The prior probability is an empirical distribution estimated from simulated data (see Section 4.1.) using a maximum likelihood estimator. The data binning technique along with the Gaussian Kernel smoother is used in order to reduce the memory footprint of the probability density function (PDF), so that the size remains constant regardless the amount of samples that have been used. The empirical prior probability was chosen instead of a conjugate prior, as the distribution may change significantly in a way that one distribution cannot describe all possible priors. The fact, that the training data are simulated provided an opportunity for a wide range of studies, such as selection of desired particle charge or tracks with desired pseudo-rapidity[2]. The Bayesian approach makes it possible to consider some patterns more likely over others, and opens new possibilities for new techniques in physics analysis.

3.1.2. Logistic regression as a background model

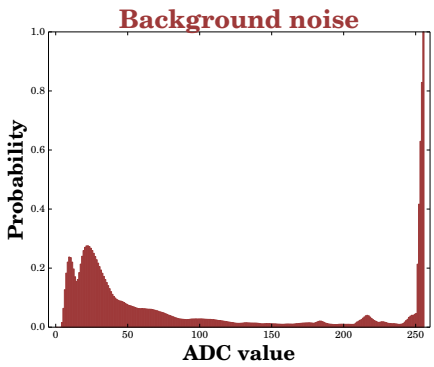

Figure 5. Scaled probability mass distribution of background noise P_N, using maximum ADC feature value of a cluster.

On the other side of the competition, the discriminative model is used to describe the background. It is a simple one-dimensional model which uses only one single feature of clusters, namely maximal value of ADC signal within a cluster. For this purpose, labeled real data has been used. The probability mass distribution can be observed on Figure 5. Note that cluster with maximum ADC equals 255 is very likely to be a noise as it indicates a saturation in the electronics. Therefore, the final posterior probability is given by

$$P(\mathbf{\Omega}|\mathbf{c}) = P_N(\max(\mathcal{R}(\mathbf{c}))), \tag{9}$$

with operator from cluster to ADC signal vector space $\mathcal{R} : C \to A$

$$\mathcal{R}(\mathbf{c}) = (\mathbf{c}_{s_1}, \ldots, \mathbf{c}_{s_n}) \tag{10}$$

[2] Pseudo-rapidity is a commonly used spatial coordinate, describing the angle of a particle relative to the beam axis.

which returns a vector of a n ADC signals of a cluster of dimension n. The $\mathbf{\Omega}$ denotes a background class and P_N is the probability mass distribution of maximum ADC value of a noise cluster. Despite of its simplicity, this noise model is seen to work quite well as a competitor to track model. It prevents attaching a cluster to a track is that does not very likely belong to any track because of its unlikely signal amplitude and therefore it reduces the number of outlier clusters on track candidates. In order to avoid overtraining, the Kolmogorov-Smirnov test is used on consecutive distribution updates until it reaches the significance level of $\alpha = 0.001$.

3.2. Measurement-to-track association

In this section a competition algorithm is described along with a decision rule, which is used in order to assign a cluster to a particular track or to background. An important issue in tracking algorithms is a way to determine number of objects as well as how to reach good computational efficiency of data association.

The computational efficiency has been accomplished using a greedy algorithm on reduced search space, using a searching cone. Because of that, global optimum is not guaranteed, however it provides good performance, as presented in a later section. Although all measurements are available at the same time, the algorithm proceeds using clusters ordered along the z coordinate which can be treated as a running time. To be precise, it processes the clusters in opposite directions to natural time elapse, namely towards the target. This approach is justified by the fact that the density of tracks drops along the detectors in the downstream direction, and therefore we start from the regions of smallest density. The reason of the lower particle trajectory density in the downstream region is the initial opening angles between tracks, accompanied by the spreading effect by the magnetic field.

The decision rule is defined in the following manner

$$\delta(\mathbf{c}) = \begin{cases} \vartheta, & \text{if } p(\mathbf{c}) \geq 1 \\ \mathbf{\Omega}, & \text{otherwise} \end{cases} \tag{11}$$

with the posterior probability ratio

$$p(\mathbf{c}) = \frac{\arg\max_{\vartheta \in T} P(\vartheta|\mathbf{c})}{P(\mathbf{\Omega}|\mathbf{c})} \tag{12}$$

where T denotes a set of track parameter vectors. The $\delta(\mathbf{c})$ function governs measurement association to a particular track/seed or to background. The latter implies a new track creation as the background cluster may belong to an other, yet undiscovered track. The initial step of track creation is called seeding: a new track object without estimated parameters is called a seed. During this stage, the seed collects all clusters within the search cone which do not belong to a track, but may share the cluster with other seeds. The overall mechanism of track competition is illustrated on Fig. 6.

As we follow the particles of interest in the detector in the upstream direction, it is possible to design a cone which will reduce the measurement search space. Therefore, a cone is defined which is governed by the following inequalities

$$tg(\beta) \geq \frac{\sqrt{(\mathbf{c}_x^n - \mathbf{c}_x)^2 + (\mathbf{c}_y^n - \mathbf{c}_y)^2}}{|\mathbf{c}_z^n - \mathbf{c}_z|} \tag{13}$$

and

$$d \geq |\mathbf{c}_z^n - \mathbf{c}_z| \tag{14}$$

where β is a cone generatrix angle, $\mathbf{c}^n$ is a last (with highest z value) measurement of a track and d is a maximum acceptable gap within a track. Collected clusters within the cone can be seen as a tree as illustrated in Fig. 7. The breadth search along with a prior probability function is used to find the most likely branch of the tree, that is to become a new track with the parameters estimated using the clusters solely from this branch. The remaining clusters are then released. The seed becomes a track candidate when the estimate of its parameters becomes possible, namely, when the following equation is satisfied

$$h \geq dim(\mathbf{p}) \tag{15}$$

where h denotes level of a seed tree. Otherwise, direct comparison of tracks is not possible. Therefore, as mention above, the seed may accept and share all clusters which were not associated with a track.

In case when (13) is satisfied, the track parameter vector is estimated for a path from a leaf (the newest cluster) until the tree root. When the parameters are estimated, the posterior probability is calculated and stored for every possible parent. Afterwards the estimated parameters from the most probable branch is chosen. A detailed data association algorithm is provided in the form of pseudo-code (Alg. 1). In the end, a second iteration over clusters is performed for reassociation of clusters improperly classified to the background using tracks found in a previous trial. In this stage, new tracks are not created.

3.3. Continuous parameters optimization

Association (see Section 3.2.) of a new cluster naturally provides additional information, so a better trajectory estimate can be achieved. Therefore, every new assignment yields parameter optimization as shown at line 10 of Algorithm 1. In order to produce a new estimate of Θ the CMA-ES [32,33] algorithm has been chosen. It is a stochastic and derivative-free evolutionary algorithm which allows the method to work whereas Quasi-Newton methods fail. The chosen approach is regarded to be robust for a non-convex functions which can comprise discontinuities, spikes or being ill-conditioned. Thus, the CMA-ES [32] in particularly is useful for solving 'black box' scenarios, where the knowledge about the underlying function is limited or an algorithm should not depend on that knowledge. The latter feature is very useful when the objective function significantly changes: severe modification to the algorithm is not needed, because of this property.

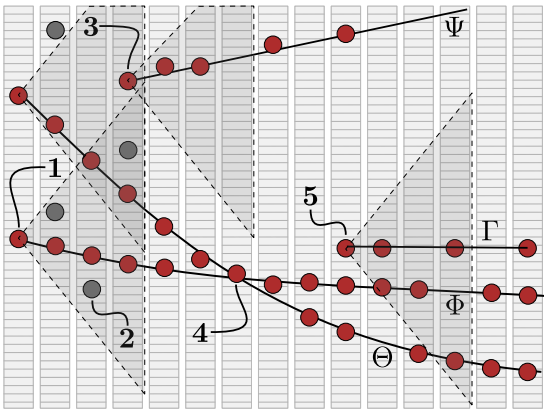

Figure 6. Illustration of the data association decisions made in a typical situation. The window slides from the left (downstream) to the right (upstream) pad-row by pad-row.
1) Failed to find a suitable track, thus a new seed is created before classification as a background; 2) $P(\Omega|\mathbf{c}) < P(\Theta|\mathbf{c})$. It is not the most probable candidate. It will become a new seed and later classified as a background; 3) $P(\Omega|\mathbf{c}) > P(\Theta|\mathbf{c})$, therefore being background is more likely, nevertheless it gets a chance as a seed, to form a track. In the end it will become a track Φ; 4) Looks for the most probable parent between Φ and Θ; 5) Being background wins, so a seed is created and it will be transformed to a track Γ.

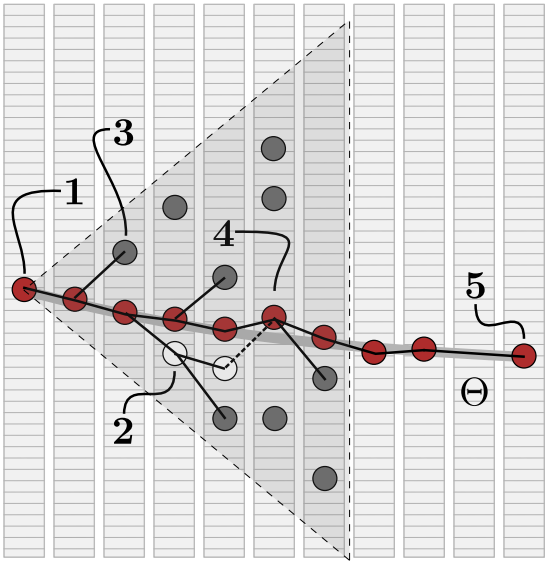

Figure 7. Illustration of the breadth search performed on a seed (within the gray cone).
1) The root of a seed tree; 2) $P(\Omega|\mathbf{c}) < P(\Theta|\mathbf{c})$. It is not the most suitable candidate. Furthermore it will not became a seed; 3) $P(\Omega|\mathbf{c}) > P(\Theta|\mathbf{c})$. Classified as a background, nevertheless it becomes as a seed first; 4) Good track cluster. 5) The track is formed, thus a breath search is not performed. A new cluster is attached to the track.

The CMA-ES consists of three main steps: (i) offspring generation – new solutions (offspring) are generated by sampling a multivariate normal distribution; (ii) selection and recombination – for each offspring an objective function is evaluated; (iii) adapting a covariance matrix and a mean; see [34] for more detailed introduction.

For the algorightm to work, only an objective function $f : \mathbb{R}^n \to \mathbb{R}$ is required to be defined.

Algorithm 1 Probabilistic data association

```
1: procedure PDA(cluster)
2:     for all tracks do                          ▷ Including mature seeds
3:         probabilities[track] ← P(track | cluster)
4:     end for
5:     bestTrack ← MAX(probabilities)
6:     if probabilities[bestTrack] <= P(Ω | cluster) then
7:         RETURN(False)          ▷ Check a premature seed or create a new seed
8:     end if
9:     ASSOCIATEWITHTRACK(bestTrack, cluster)
10:    OPTIMIZEPARAMETERS(bestTrack)
11:    RETURN(True)
12: end procedure
```

The evolutionary tracker uses χ^2 as an objective function, defined in the following manner

$$\chi^2 = \sum_{i=1}^{n} \chi_i^2, \tag{16}$$

with

$$\chi_i^2 = \mathbf{d} \cdot diag\{\mathbf{d}\} \cdot \Sigma^{-1}, \tag{17}$$

and distance vector

$$\mathbf{d} = \mathcal{S}(r(\hat{\Theta}, c_z^i)) - c_z^i \tag{18}$$

where $\mathbf{c}_z^i$ is an i-th (out of n) cluster of a track Θ and $r(\cdot, \cdot)$ stands for an extrapolation function (3).

Given that event samples of several 100 millions need to be reconstructed, the performance of event reconstruction algorithm is a very relevant issue. Therefore we used an improved method of CMA-ES called Active-CMA-ES (aCMA-ES) [35]. The aCMA-ES differs from the original method by using information from unsuccessful offspring in addition. The information is used in order to reduce variance in unpromising directions. The authors demonstrate that an algorithm is superior in performance compared to Original-CMA-ES [32] as well as to Hybrid-CMA-ES. The highest performance, more than 40%, can be observed in case when the eigenvalue of a particular objective function dominates the others.

4. Results

In this section we briefly review the methodology of preparing the input test data as well as the technique to check the correctness of reconstruction. Finally, the perfor-

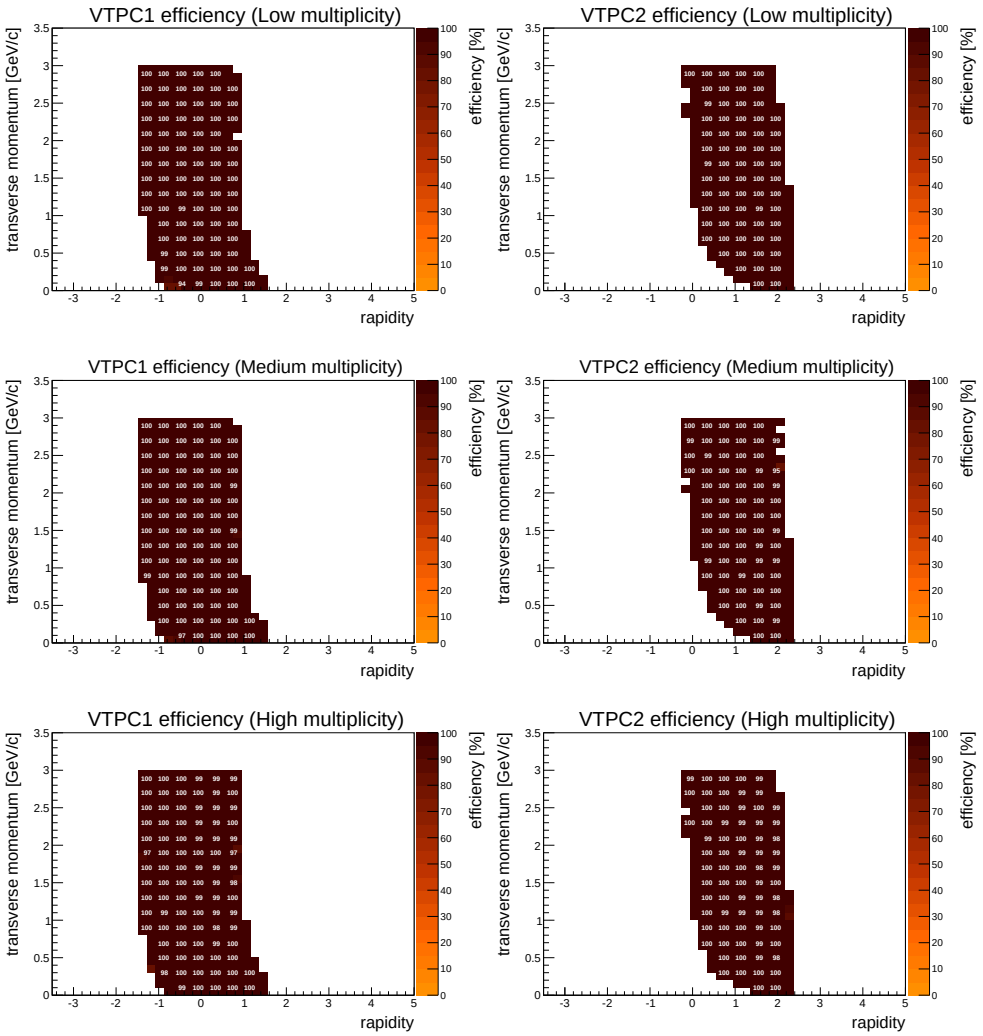

Figure 8. Efficiency for VTPC1 (top) and VTPC2 (bottom) chamber. Multiplicity denotes the number of particles in the studied region of rapidity region. Low, medium and high multiplicity are respectively 11.5, 115.6, 423.1 particles per event per unit rapidity.

Table 1. Summary of fake track rates [%].

TPC	Low multiplicity	Medium multiplicity	High multiplicity
VTPC 1	0.0084	0.0042	0.0018
VTPC 2	0.0077	0.0038	0.0033

mance results will be shown.

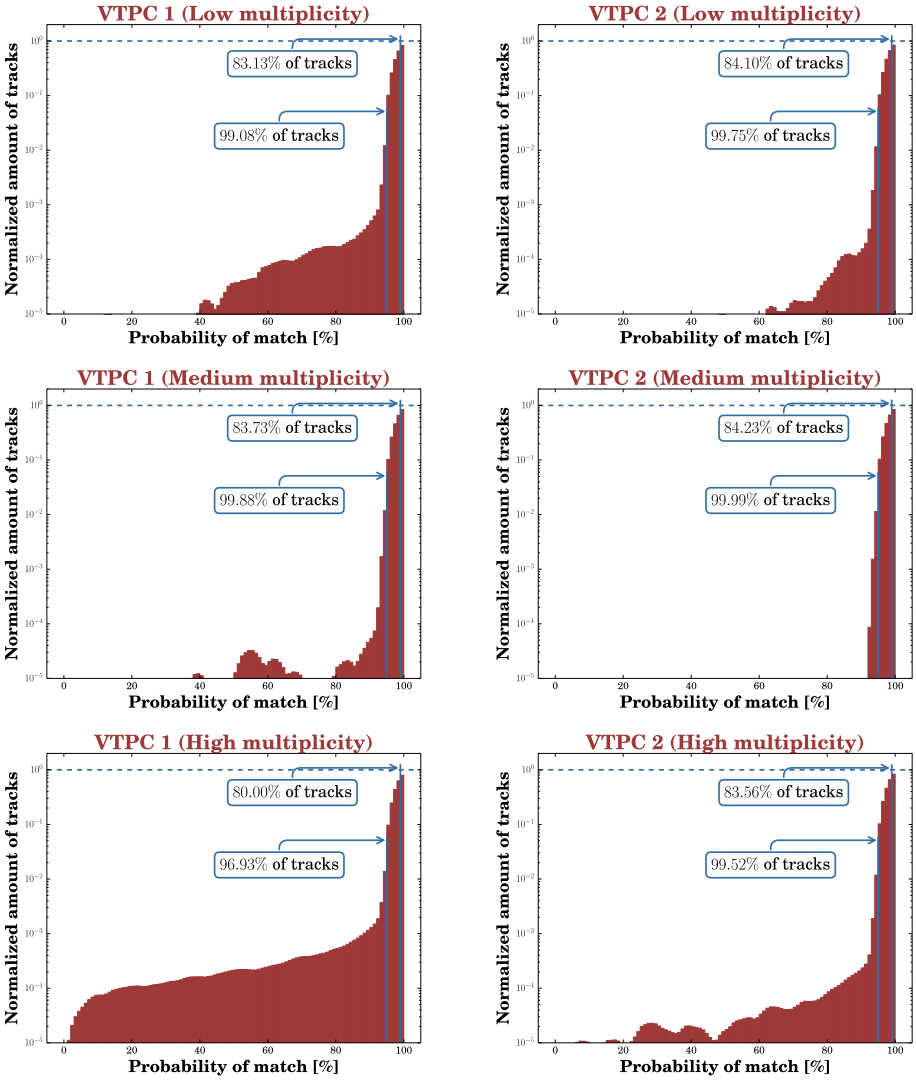

Figure 9. Goodness of fit using Pearson's χ^2 test for VTPC1 (top) and VTPC2 (bottom) chamber. Multiplicity denotes the number of particles per event per unit rapidity (often denoted by $\frac{dn}{dy}$). Low, medium and high multiplicity are respectively 11.5, 115.6, 423.1.

4.1. Simulation of the events

In order to study the efficiency and correctness of the algorithm, simulated data where prepared. For event generating purpose, the flat phase space generator was used, which produces necessary input for the GEANT toolkit [36]. The next step is

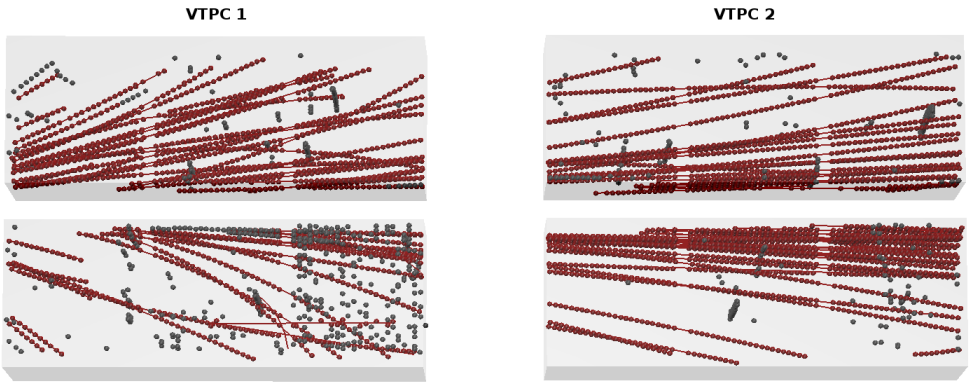

Figure 10. An example reconstructed event of medium multiplicity. Black points denotes those classified as background.

to produce events using the reconstruction software. In order to do that, an electronic response of the TPCs readout was simulated, resulting in reference raw data. Real detector noise clusters were superimposed on the simulated raw data in order to imitate real conditions. Starting from this point, the same reconstruction software is used on simulated as on experimental data. At this step the reconstructed simulated data can be compared to the simulated event data, as shown in Fig. 10.

4.2. Matching procedure

The study of the new algorithm is based on a comparison of results between reconstructed tracks to the simulated tracks from which they originate. The procedure is called matching and it is done by a correlation between clusters and simulated points. In order to make a decision on whether a cluster belongs to a simulated track point, a maximum distance is defined. We use a distance of 0.25 cm, which is large enough to ensure matching despite possible small distortion effects by detector effects. In this procedure, multiple matches may occur. The ambiguities are handled on the match analysis level.

4.3. Reconstruction performance

Until now, the method of generating input data as well as the matching procedure were described. In this part we discuss the performance of the algorithm implemented using the SHINE Offline framework [37].

The results of the reconstruction efficiency study is shown in Fig. 8. The rapidity parameter here denotes rapidity understood in the center of mass system with the

assumption of pion mass for the produced particle. As mentioned previously, this algorithm is applied only for the VTPC detectors, immersed in a magnetic field. The shape of histograms is determined by the detector acceptance, therefore it contributes to a significant efficiency drop at the edges. Furthermore, for high multiplicity events efficiency losses occur due to the finite two-track resolution of the detector. These situations can result in merged tracks, which contribute to the fake track rate. The fake track rates shown in Tab. 1, indicate that the presented method is also robust in terms of forming fake tracks from noise clusters. Within the detector acceptance, the algorithm is seen to perform remarkably well, especially that pure local tracking was considered to be unachievable in chambers with the highest track density [38].

5. Conclusions and future work

The results show that the proposed algorithm can reliably track an unknown number of particles traversing the detector, including those sharing parts of their trajectories. This was achieved using simple Bayes models without incorporating the Kalman filter. Notably, the reconstruction performance is relatively high given the small amount of measurement information provided by the detector. Therefore, it is expected to have a large potential in the field of event reconstruction. The method presents an applicability for problems where the equations of motion are non-linear, non-convex functions with discontinuities. However, the algorithm shows sensitivity to chosen parameters in terms of computational time. This motivates further studies on algorithm performance as a function of algorithm parameters, along with the exact implementation of the gradient function. After the optimal algorithmic parameters are determined, a full comparison with the Kalman filter is to be performed. Furthermore, track and background clusters are to be analyzed in order to find better models, emphasizing the latter one. A better model of background should decrease number of outlier clusters and would lead to even higher reconstruction efficiency and goodness of fit.

Presented algorithm is developed as a part of NA61/SHINE data reconstruction software, performing a subdetector wise reconstruction.

Acknowledgements

The author would like to thank Przemysław Spurek and András László whose comments have improved the quality of this paper as well as to Roman Płaneta for his support. Great thanks to Balázs Kégl, Péter Köversárki and Darko Veberič for fruitful discussions on possible solutions to the presented problem. Last but not least, many

thanks to the NA61/SHINE collaboration and CERN doctoral student program for supporting this work. This work was supported by the National Science Centre of Poland, Grant No. UMO-2012/04/M/ST2/00816.

6. References

[1] Mankel R., *Pattern recognition and event reconstruction in particle physics experiments.* Reports on Progress in Physics, 2004, 67, pp. 553–622.

[2] Billoir P., *Track Fitting With Multiple Scattering: A New Method.* Nucl. Instrum. Meth., 1984, A225, pp. 352–366.

[3] Billoir P., Fruhwirth R., Regler M., *Track Element Merging Strategy and Vertex Fitting in Complex Modular Detectors.* Nucl. Instrum. Meth., 1985, A241, pp. 115–131.

[4] Billoir P., *Progressive track recognition with a Kalman like fitting procedure.* Comput. Phys. Commun., 1989, 57, pp. 390–394.

[5] Fruhwirth R., *Application of Kalman filtering to track and vertex fitting.* Nucl. Instrum. Meth., 1987, A262, pp. 444–450.

[6] Kalman R.E., *A new approach to linear filtering and prediction problems.* Transactions of the ASME–Journal of Basic Engineering, 1960, 82(Series D), pp. 35–45.

[7] Myers S., *The large hadron collider 2008-2013.* International Journal of Modern Physics A, 2013, 28(25), pp. 1330035.

[8] Cornelissen T., Elsing M., Fleischmann S., Liebig W., Moyse E., Salzburger A., *Concepts, Design and Implementation of the ATLAS New Tracking (NEWT).* Technical Report ATL-SOFT-PUB-2007-007. ATL-COM-SOFT-2007-002, CERN, 2007.

[9] Collaboration A., Airapetian A., Cindro V., Filipčič A., Kramberger G., Mandić I., Mikuž M., Tadel M., Žontar D., *ATLAS detector and physics performance.* Technical design report. ATLAS, 1999.

[10] Palmonari F., *CMS tracker performance.* Nucl.Instrum.Meth., 2013, A699, pp. 144–148.

[11] Merkel P., *CMS tracker performance.* Nucl.Instrum.Meth., 2013, A718, pp. 339–341.

[12] CMS collaboration and others, *Description and performance of track and primary-vertex reconstruction with the cms tracker.* Journal of Instrumentation, 2014, 9(10), pp. P10009.

[13] Abelev B., et al., *Upgrade of the ALICE Experiment: Letter Of Intent.* J. Phys., 2014, G41, pp. 087001.

[14] Aamodt K., et al., *The ALICE experiment at the CERN LHC.* JINST, 2008, 3, pp. S08002.

[15] Amoraal J., Collaboration L., et al., *Alignment of the LHCb detector with kalman filter fitted tracks.* In: *Journal of Physics: Conference Series.* vol. 219., IOP Publishing, 2010, pp. 032028.

[16] Rodrigues E., *The LHCb track kalman fit.* Note LHCb-2007-014, 2007, 164.

[17] Hernando J., Rodrigues E., *Tracking event model, LHCb internal note, LHCb-2007-007.* CERN-LHCb-2007-007.

[18] Schiller M., *Standalone track reconstruction for the Outer Tracker of the LHCb experiment using a cellular automaton.* PhD thesis, Uni Heidelberg 2007.

[19] Passaleva G., *A recurrent neural network for track reconstruction in the LHCb muon system.* In: *Nuclear Science Symposium Conference Record, 2008. NSS'08. IEEE*, IEEE, 2008, pp. 867–872.

[20] Pulvirenti A., Badala A., Barbera R., Lo Re G., Palmeri A., et al., *Neural tracking in the ALICE Inner Tracking System.* Nucl. Instrum. Meth., 2004, A533, pp. 543–559.

[21] Badala A., Barbera R., Lo Re G., Palmeri A., Pappalardo G., et al., *Combined tracking in the ALICE decetctor.* Nucl. Instrum. Meth., 2004, A534, pp. 211–216.

[22] Badala A., Barbera R., Re G.L., Palmeri A., Pappalardo G., Pulvirenti A., Riggi F., *Neural tracking in alice.* Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 2003, 502(2), pp. 503–506.

[23] Strandlie A., Frühwirth R., *Track and vertex reconstruction: From classical to adaptive methods.* Reviews of Modern Physics, 2010, 82(2), pp. 1419.

[24] Abgrall N., et al., *Na61/shine facility at the cern sps: beams and detector system.* Journal of Instrumentation, 2014, 9(06), pp. P06005.

[25] Nygren D.R., *Proposal to investigate the feasibility of a novel concept in particle detection.* LBL Internal Report, 1974.

[26] Wyszynski O., *Trigger system of the NA61/SHINE experiment at the CERN SPS*, 2014.

[27] Laszlo A., Denes E., Fodor Z., Kiss T., Kleinfelder S., Soos C., Tefelski D., Tolyhi T., Vesztergombi G., Wyszynski O., *Design and performance of the data acquisition system for the na61/shine experiment at cern.* arXiv preprint arXiv:1505.01004, 2015.

[28] Gorbunov S., Kisel I., *Analytic formula for track extrapolation in non-homogeneous magnetic field.* Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 2006, 559(1), pp. 148–152.

[29] Jordan A., *On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes.* Advances in neural information processing systems, 2002, 14, pp. 841.

[30] Hand D.J., Yu K., *Idiot's bayesnot so stupid after all?* International statistical review, 2001, 69(3), pp. 385–398.

[31] Domingos P., Pazzani M., *Beyond independence: Conditions for the optimality of the simple bayesian classifier.* In: *Machine Learning*, Morgan Kaufmann, 1996, pp. 105–112.

[32] Hansen N., Ostermeier A., *Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation.* In: *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation.* IEEE 1996 pp. 312–317.

[33] Hansen N., Ostermeier A., *Completely derandomized self-adaptation in evolution strategies.* Evolutionary Computation, 2001, 9(2), pp. 159–195.

[34] Hansen N., *The CMA evolution strategy: a comparing review.* In Lozano J., Larranaga P., Inza I., Bengoetxea E., eds.: *Towards a new evolutionary computation. Advances on estimation of distribution algorithms.* Springer 2006 pp. 75–102.

[35] Jastrebski G.A., Arnold D.V., *Improving evolution strategies through active covariance matrix adaptation.* In: *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on.* IEEE 2006 pp. 2814–2821.

[36] Agostinelli S.e.a., *GEANT4: A simulation toolkit.* Nuclear Instruments and Methods in Physics Research, 2003, A506, pp. 250–303.

[37] Sipos R., Laszlo A., Marcinek A., Paul T., Szuba M., Unger M., Veberic D., Wyszynski O., *The offline software framework of the na61/shine experiment.* In: *Journal of Physics: Conference Series.* vol. 396., IOP Publishing, 2012, pp. 022045.

[38] Irmscher D., *Philosophy and parts of the global tracking chain.* NA49 Note number 131 (1997).